Exhibit H-1

Commissioner for Patents
P.O. Box 1450,
Alexandria, VA 22313-1450

US Patent: 5,819,292 By: RLY:rac
Issued: October 6, 1998
DLAP File No.: 347155-29

Title: **METHOD FOR MAINTAINING CONSISTENT STATES OF A FILE SYSTEM AND FOR CREATING USER-ACCESSIBLE READ-ONLY COPIES OF A FILE SYSTEM**

Mail Date: October 25, 2007 Due Date: N/A
The following have been received in the U.S. Patent and Trademark Office on the date stamped hereon:

1. Transmittal Form;
2. Request for Ex Parte Reexamination Transmittal Form;
3. Attachment to Request for Re-Examination (Form PTO-1465) Providing Information on U.S. Patent No. 5,819,292; and
4. Information Disclosure Statement & PTO-1449; w/6 references;
5. Certificate of Mailing By Express Mail No.: EV 978 428 074 US; and
6. Return Post Card.

PTO/SB/21 (10-07)
Approved for use through 10/31/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

U.S. Patent No.	5,819,292
Filing Date	May 31, 1995
First Named Inventor	Hitz et al.
Art Unit	N/A
Examiner Name	N/A
Attorney Docket Number	347155-29

Total Number of Pages in This Submission: 238

ENCLOSURES *(Check all that apply)*

- [ ] Fee Transmittal Form
  - [ ] Fee Attached
- [ ] Amendment/Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [x] Information Disclosure Statement & PTO-1449
- [ ] Certified Copy of Priority Document(s)
- [ ] Reply to Missing Parts/ Incomplete Application
  - [ ] Reply to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [x] Request for Ex Parte Reexamination Transmittal Form
- [ ] CD, Number of CD(s) ______
  - [ ] Landscape Table on CD
- [ ] After Allowance Communication to TC
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) (please Identify below):
  1. Attachment to Request for Re-Examination (Form PTO-1465) Providing Information on U.S. Patent No. 5,819,292; and
  2. Return Post Card.

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	DLA Piper US LLP
Signature	Ronald L. Yin
Printed name	Ronald L. Yin
Date	October 25, 2007
Reg. No.	27,607

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage via "EXPRESS MAIL No. EV 978 428 074 US in an envelope addressed to: Commissioner for Patents, MS BOX REEXAM, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature	Ronald L. Yin
Typed or printed name	Ronald L. Yin
Date	October 25, 2007

Attorney Docket No. 347155-29

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

EXPRESS MAIL NUMBER: EV 978 428 074 US DATE OF DEPOSIT: October 25, 2007

I hereby certify that this paper is being deposited with the United States Postal Service "EXPRESS MAIL Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to: Commissioner for Patents; P.O. Box 1450, Alexandria, VA 22313-1450.

Rosa A. Caviedes

Rosa A. Caviedes

* * *

CERTIFICATE OF MAILING BY EXPRESS MAIL

COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, VA 22313-14500

Sir:

Transmitted herewith for filing is the following:

1. Transmittal Form;
2. Request for Ex Parte Reexamination Transmittal Form (+1 copy);
3. Attachment to Request for Re-Examination (Form PTO-1465) Providing Information on U.S. Patent No. 5,819,292;
4. Information Disclosure Statement & PTO-1449 w/6 references;
5. Certificate of Mailing By Express Mail No.: EV 978 428 074 US; and
6. Return Post Card.

EV978428074US

PTO/SB/57 (09-07)
Approved for use through 08/31/2010. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

(Also referred to as FORM PTO-1465)

REQUEST FOR *EX PARTE* REEXAMINATION TRANSMITTAL FORM

Address to:
Mail Stop *Ex Parte* Reexam
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Attorney Docket No.:

Date:

1. [X] This is a request for *ex parte* reexamination pursuant to 37 CFR 1.510 of patent number 5,819,292 issued October 6, 1998. The request is made by:

   [ ] patent owner. [X] third party requester.

2. [X] The name and address of the person requesting reexamination is:

   Ronald L. Yin
   DLA Piper US LLP
   2000 University Avenue, East Palo Alto, CA 94303

3. [ ] a A check in the amount of $________ is enclosed to cover the reexamination fee, 37 CFR 1.20(c)(1),

   [X] b. The Director is hereby authorized to charge the fee as set forth in 37 CFR 1.20(c)(1) to Deposit Account No. 07-1896 (submit duplicative copy for fee processing); or

   [ ] c. Payment by credit card. Form PTO-2038 is attached.

4. [X] Any refund should be made by [ ] check or [X] credit to Deposit Account No. 07-1896. 37 CFR 1.26(c). If payment is made by credit card, refund must be to credit card account.

5. [X] A copy of the patent to be reexamined having a double column format on one side of a separate paper is enclosed. 37 CFR 1.510(b)(4)

6. [ ] CD-ROM or CD-R in duplicate, Computer Program (Appendix) or large table
   [ ] Landscape Table on CD

7. [ ] Nucleotide and/or Amino Acid Sequence Submission
   If applicable, items a. – c. are required.

   a. [ ] Computer Readable Form (CRF)
   b. Specification Sequence Listing on:
      i. [ ] CD-ROM (2 copies) or CD-R (2 copies); **or**
      ii. [ ] paper
   c. [ ] Statements verifying identity of above copies

8. [X] A copy of any disclaimer, certificate of correction or reexamination certificate issued in the patent is included.

9. [X] Reexamination of claim(s) 1-4, 8-20 is requested.

10. [X] A copy of every patent or printed publication relied upon is submitted herewith including a listing thereof on Form PTO/SB/08, PTO-1449, or equivalent.

11. [ ] An English language translation of all necessary and pertinent non-English language patents and/or printed publications is included.

PTO/SB/57 (09-07)
Approved for use through 08/31/2010. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

12. [X] The attached detailed request includes at least the following items:

a. A statement identifying each substantial new question of patentability based on prior patents and printed publications. 37 CFR 1.510(b)(1)
b. An identification of every claim for which reexamination is requested, and a detailed explanation of the pertinency and manner of applying the cited art to every claim for which reexamination is requested. 37 CFR 1.510(b)(2)

13. [ ] A proposed amendment is included (only where the patent owner is the requester). 37 CFR 1.510(e)

14. [X] a. It is certified that a copy of this request (if filed by other than the patent owner) has been served in its entirety on the patent owner as provided in 37 CFR 1.33(c).
The name and address of the party served and the date of service are:

Steven A. Swernofsky
P.O. Box 640640
San Jose, CA 95164-0640

Date of Service: October 25, 2007; or

[ ] b. A duplicate copy is enclosed since service on patent owner was not possible.

15. Correspondence Address: Direct all communication about the reexamination to:

[X] The address associated with Customer Number: 26379

OR

[ ] Firm or Individual Name

Address		
City	State	Zip
Country		
Telephone	Email	

16. [X] The patent is currently the subject of the following concurrent proceeding(s):
[ ] a. Copending reissue Application No. ________.
[ ] b. Copending reexamination Control No. ________.
[ ] c. Copending Interference No. ________.
[X] d. Copending litigation styled:

Network Appliance, Inc. v. SUN Microsystems, Inc.
9:07CV206, USDC ED Tx. (Lufkin Div.)

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Ronald L. Yin
Authorized Signature

October 25, 2007
Date

Ronald L. Yin
Typed/Printed Name

27,607
Registration No.

[ ] For Patent Owner Requester
[X] For Third Party Requester

PTO/SB/57 (09-07)
Approved for use through 08/31/2010. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

(Also referred to as FORM PTO-1465)

REQUEST FOR *EX PARTE* REEXAMINATION TRANSMITTAL FORM

Address to:
Mail Stop *Ex Parte* Reexam
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Attorney Docket No.:

Date:

1. [X] This is a request for *ex parte* reexamination pursuant to 37 CFR 1.510 of patent number 5,819,292 issued October 6, 1998. The request is made by:

   [ ] patent owner. [X] third party requester.

2. [X] The name and address of the person requesting reexamination is:

   Ronald L. Yin
   DLA Piper US LLP
   2000 University Avenue, East Palo Alto, CA 94303

3. [ ] a. A check in the amount of $______ is enclosed to cover the reexamination fee, 37 CFR 1.20(c)(1);

   [X] b. The Director is hereby authorized to charge the fee as set forth in 37 CFR 1.20(c)(1) to Deposit Account No. 07-1896 (submit duplicative copy for fee processing); or

   [ ] c. Payment by credit card. Form PTO-2038 is attached.

4. [X] Any refund should be made by [ ] check or [X] credit to Deposit Account No. 07-1896. 37 CFR 1.26(c). If payment is made by credit card, refund must be to credit card account.

5. [X] A copy of the patent to be reexamined having a double column format on one side of a separate paper is enclosed. 37 CFR 1.510(b)(4)

6. [ ] CD-ROM or CD-R in duplicate, Computer Program (Appendix) or large table
   [ ] Landscape Table on CD

7. [ ] Nucleotide and/or Amino Acid Sequence Submission
   If applicable, items a. – c. are required.

   a. [ ] Computer Readable Form (CRF)
   b. Specification Sequence Listing on:
      i. [ ] CD-ROM (2 copies) or CD-R (2 copies); **or**
      ii. [ ] paper
   c. [ ] Statements verifying identity of above copies

8. [X] A copy of any disclaimer, certificate of correction or reexamination certificate issued in the patent is included.

9. [X] Reexamination of claim(s) 1-4, 8-20 is requested.

10. [X] A copy of every patent or printed publication relied upon is submitted herewith including a listing thereof on Form PTO/SB/08, PTO-1449, or equivalent.

11. [ ] An English language translation of all necessary and pertinent non-English language patents and/or printed publications is included.

PTO/SB/57 (09-07)
Approved for use through 08/31/2010. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

12. [X] The attached detailed request includes at least the following items:

a. A statement identifying each substantial new question of patentability based on prior patents and printed publications. 37 CFR 1.510(b)(1)
b. An identification of every claim for which reexamination is requested, and a detailed explanation of the pertinency and manner of applying the cited art to every claim for which reexamination is requested. 37 CFR 1.510(b)(2)

13. [ ] A proposed amendment is included (only where the patent owner is the requester). 37 CFR 1.510(e)

14. [X] a. It is certified that a copy of this request (if filed by other than the patent owner) has been served in its entirety on the patent owner as provided in 37 CFR 1.33(c).
The name and address of the party served and the date of service are:

Steven A. Swernofsky
P.O. Box 640640
San Jose, CA 95164-0640

Date of Service: October 25, 2007 ; or

[ ] b. A duplicate copy is enclosed since service on patent owner was not possible.

15. Correspondence Address: Direct all communication about the reexamination to:

[X] The address associated with Customer Number: 26379

OR

[ ] Firm or Individual Name

Address			
City		State	Zip
Country			
Telephone		Email	

16. [X] The patent is currently the subject of the following concurrent proceeding(s):
[ ] a. Copending reissue Application No. ______.
[ ] b. Copending reexamination Control No. ______.
[ ] c. Copending Interference No. ______.
[X] d. Copending litigation styled:

Network Appliance, Inc. v. SUN Microsystems, Inc.
9:07CV206, USDC ED Tx. (Lufkin Div.)

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Ronald L. Yin
Authorized Signature

October 25, 2007
Date

Ronald L. Yin
Typed/Printed Name

27,607
Registration No.

[ ] For Patent Owner Requester
[X] For Third Party Requester

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Hitz et al.

U.S. Patent No. 5,819,292 Issued: October 6, 1998

Filed: May 31, 1995

Docket No.: 347155-29

Title: METHOD FOR MAINTAINING CONSISTENT STATES OF A FILE SYSTEM AND FOR CREATING USER-ACCESSIBLE READ-ONLY COPIES OF A FILE SYSTEM

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage Via Express Mail No. EV 978 428 074 US in an envelope addressed to: Commissioner of Patents, MS BOX REEXAM, P.O. Box 1450, Alexandria, VA 22313-1450, on:
October 25, 2007

Ronald L. Yin

Ronald L. Yin

* * *

ATTACHMENT TO REQUEST FOR RE-EXAMINATION (FORM PTO-1465) PROVIDING INFORMATION ON U.S. PATENT NO. 5,819,292

ATTN: BOX REEXAM
Commissioner of Patents
Washington, D.C. 20231

Sir:

Pursuant to 35 U.S.C. §§ 302-307 and 37 CFR § 1.510, this is a request for ex-parte reexamination of United States Patent Number 5,819,292 which issued on October 6, 1998 to Hitz et al. (the "'292 Patent").

I. CLAIMS FOR WHICH REEXAMINATION IS REQUESTED

Reexamination is requested of Claims 1-4, and 8-20 of the '292 Patent in view of the prior art listed on the Citation of Prior Art under 37 CFR § 1.501 and 35 U.S.C. § 301 which is submitted with the Request for Reexamination.

II. EXPLANATION OF PERTINENCY AND MANNER OF APPLYING CITED PRIOR ART TO EVERY CLAIM FOR WHICH REEXAMINATION IS REQUESTED

Introduction

One or more prior art references submitted with the Citation of Prior Art render the claims of the '292 Patent either anticipated under 35 USC 102 (a), (b) or (e) or unpatentable under 35 USC 103 so that substantial new questions of patentability of the claims in the '292 Patent have been raised by this request for reexamination. For each claim for which reexamination is sought, a specific citation of the prior art or combination of prior art pertinent to the claim and a description of the relevancy of that prior art to the claim are set forth below in more detail. None of the prior art cited in the Citation of Prior Art was submitted to the examiner or considered by the examiner during the prosecution of the '292 Patent.

Invalidity

Claim 1

Claim 1 of the '292 Patent is remarkably similar if not identical to claim 52 as originally filed in the application which matured into USP 7,174,352 ("'352 Patent") (see Ex. A attached hereto). The '352 Patent claims priority from the filing date of the '292 Patent and is alleged to be a continuation-in-part of the '292 Patent. A comparison of issued claim 1 of the '292 Patent to originally filed claim 52 of the '352 shows that the claims are virtually identical.

US Patent No. 5,819,292	Originally Filed Claim 52 of '352 Patent
1. A method for recording a plurality of data about a plurality of blocks of data stored in storage means comprising the steps of:	52. A method for recording a plurality of data about a plurality of blocks of data stored in storage means, comprising the steps of:
maintaining a means for recording multiple usage bits per block of said storage means,	maintaining a means for recording multiple usage bits per block of said storage means; and
storing, in said means for recording multiple usage bits per block, multiple bits for each of said plurality of said blocks of said storage means; and	storing in said means for recording multiple usage bits per block, multiple bits for each of said plurality of said blocks of said storage means,
reusing at least one of said plurality of blocks of data in response to at least one of said multiple usage bits.	at least one of said multiple bits being indicative of block reusability.

As can be seen, the preamble and the first two elements of claim 1 of the '292 Patent and originally filed claim 52 of the '352 Patent are word-for-word identical. As for the last element of claim 1 of the '292 Patent, claim 52 also teaches the reusability concept, although not expressed as a separate method step. Originally filed claim 52 of the '352 Patent was rejected by examiner Le (different from the examiner for the '292 Patent), who rejected claim 52 as being anticipated under 35 U.S.C. 102(b) by Lorie. (see Ex. B). In rejecting claim 52, the examiner noted that Lorie disclosed Mod bit and shadow bit (pages 95-97) which met the claimed limitation of "maintaining a means for recording multiple usage bits per block of said storage means." Further, Lorie disclosed using shadow bits to release slots in the current map (page 99 of Lorie) as meeting the claimed limitation of "storing in said means for recording multiple usage bits per block, multiple bits for each of said plurality of said blocks of said storage means, at least one of said multiple bits being indicative of block reusability" (see page 3 of Ex. B)

The applicants conceded the relevance of Lorie and the appropriateness of the rejection when claim 52 was cancelled in response to the rejection. (see Ex. C).

Thus, claim 1 of the '292 Patent is anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
1. A method for recording a plurality of data about a plurality of blocks of data stored in storage means comprising the steps of:	Lorie discloses storing a plurality of states. See at least the abstract.
maintaining a means for recording multiple usage bits per block of said storage means,	Mod bit and shadow bits (page 95-97) record multiple usage bits per block.
storing, in said means for recording multiple usage bits per block, multiple bits for each of said plurality of said blocks of said storage means; and	Shadow bits release slots in the current bit map (page 99) demonstrates reusability.
reusing at least one of said plurality of blocks of data in response to at least one of said multiple usage bits.	Actual release of a slot by a shadow bit results in the reusing of one of the blocks (page 99).

Claim 1 of the '292 Patent is also anticipated by U.S. Pat 5,129,085 ("Yamasaki") as follows:

US Patent No. 5,819,292	U.S. patent 5,129,085 ("Yamasaki")
1. A method for recording a plurality of data about a plurality of blocks of data stored in storage means comprising the steps of:	Yamasaki teaches subdividing a memory into blocks and using two bitmaps to store information about the usage of each block. See

US Patent No. 5,819,292	U.S. patent 5,129,085 ("Yamasaki")
	Figure 1.
maintaining a means for recording multiple usage bits per block of said storage means,	See column 2, lines 54-66 which discloses a first bit map and a second bit map corresponding to a delimited memory area.
storing, in said means for recording multiple usage bits per block, multiple bits for each of said plurality of said blocks of said storage means; and	The two bit maps store multiple usage bits per delimited memory block (see column 2, lines 54-66).
reusing at least one of said plurality of blocks of data in response to at least one of said multiple usage bits.	If the bit in the second map is not written, the corresponding memory block is released; see column 4, lines 31-36.

Claim 2

Claim 2 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
2. A method for maintaining a file system stored in non-volatile storage means at successive consistency points said file system comprising blocks of data, said blocks of data comprising blocks of regular file data and blocks of meta-data file data referencing said blocks of data of said file system, said meta file data comprising a file system information structure comprising data describing said file system at a first consistency point said computer system further comprising memory means, said method comprising the steps of:	Section 4.3 and Figure 2 of Borg discloses a file stored in a non-volatile means (a disk) blocks of regular file data and blocks of meta-data file referencing the blocks of data of the file system, at a first consistency point (see Figure 2a). The computer system also has memory means (see Figure 2a).
maintaining a plurality of modified blocks of regular file data and meta-data file data in said memory means, said modified blocks of data comprising blocks of data modified from said first consistency point;	Figure 2b shows the modified blocks of regular file data and meta-data file data being maintained in memory. The modified blocks of data are modified from the first consistency point shown in Figure 2a.
designating as dirty blocks of meta-data file data referencing said modified blocks of regular file data and meta-data file data, said dirty blocks of meta-data file data comprising blocks of meta-data file data to be included in a second consistency point;	See Figures 2(a-c) and discussion in section 4.3
copying said modified blocks of regular file data referenced by said dirty blocks of meta-	See Figures 2(a-c) and discussion in section 4.3

US Patent No. 5,819,292	Borg
data file data to free blocks of said non-volatile storage means;	
copying blocks comprising said modified blocks of meta-data file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;	See Figures 2(a-c) and discussion in section 4.3
modifying a copy of said file system information structure maintained in said memory means to reference said dirty blocks of meta-data file data:	See Figures 2(a-d) and discussion in section 4.3
copying said modified file system information structure to said non-volatile storage means.	See Figures 2(a-d) and discussion in section 4.3

Claim 2 of the '292 Patent is also anticipated by Rosenblum II, as follows:

US Patent No. 5,819,292	Rosenblum II
2. A method for maintaining a file system stored in non-volatile storage means at successive consistency points said file system comprising blocks of data, said blocks of data comprising blocks of regular file data and blocks of meta-data file data referencing said blocks of data of said file system, said meta file data comprising a file system information structure comprising data describing said file system at a first consistency point said computer system further comprising memory means, said method comprising the steps of:	See sections 4.4.1 (page 9) with regard to the discussion on the maintenance of plural consistency points. See also section 2.2 (page 2) and section 4.4.1 (page 9) with regard to providing a memory (or cache).
maintaining a plurality of modified blocks of regular file data and meta-data file data in said memory means, said modified blocks of data comprising blocks of data modified from said first consistency point;	See section 4 in general and section 4.4.1 in particular. See also section 4.2.1.
designating as dirty blocks of meta-data file data referencing said modified blocks of regular file data and meta-data file data, said dirty blocks of meta-data file data comprising blocks of meta-data file data to be included in a second consistency point;	See section 4 in general and section 4.4.1 in particular. See also section 4.2.1.
copying said modified blocks of regular file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;	See section 4 in general and section 4.4.1 in particular. See also section 4.2.1.
copying blocks comprising said modified	See section 4 in general and section 4.4.1 in

US Patent No. 5,819,292	Rosenblum II
blocks of meta-data file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;	particular. See also section 4.2.1.
modifying a copy of said file system information structure maintained in said memory means to reference said dirty blocks of meta-data file data:	See section 4 in general and section 4.4.1 in particular. See also section 4.2.1.
copying said modified file system information structure to said non-volatile storage means.	See section 4 in general and section 4.4.1 in particular. See also section 4.2.1.

Claim 2 of the '292 Patent is also anticipated by Hecht, as follows:

US Patent No. 5,819,292	Hecht
2. A method for maintaining a file system stored in non-volatile storage means at successive consistency points said file system comprising blocks of data, said blocks of data comprising blocks of regular file data and blocks of meta-data file data referencing said blocks of data of said file system, said meta file data comprising a file system information structure comprising data describing said file system at a first consistency point said computer system further comprising memory means, said method comprising the steps of:	Section 2 and in particular sections 2.2 and 2.3 discloses storing of both the old page and modified page in disk, written thereto from memory. See also section 3 with regard to the disclosure of a main memory.
maintaining a plurality of modified blocks of regular file data and meta-data file data in said memory means, said modified blocks of data comprising blocks of data modified from said first consistency point;	Section 2 and in particular sections 2.2 and 2.3.
designating as dirty blocks of meta-data file data referencing said modified blocks of regular file data and meta-data file data, said dirty blocks of meta-data file data comprising blocks of meta-data file data to be included in a second consistency point;	Section 2 and in particular sections 2.2 and 2.3.
copying said modified blocks of regular file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;	Section 2 and in particular sections 2.2 and 2.3.
copying blocks comprising said modified blocks of meta-data file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;	Section 2 and in particular sections 2.2 and 2.3.

US Patent No. 5,819,292	Hecht
modifying a copy of said file system information structure maintained in said memory means to reference said dirty blocks of meta-data file data:	Section 2 and in particular sections 2.2 and 2.3.
copying said modified file system information structure to said non-volatile storage means.	Section 2 and in particular sections 2.2 and 2.3.

Claim 3

Claim 3 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
3. The method of claim 2 wherein said blocks of meta-file data comprise one or more blocks of inode file data and one or more blocks of blockmap file data and wherein said step of copying said modified blocks of meta-data file data to free blocks of said non-volatile storage means further comprises the steps of:	An inode file data is disclosed in section 4.2.
copying an inode referencing one or more blocks of blockmap file data to a block of inode file data maintained in said memory means;	The inode file data is modified in memory – see sections 4.2, 4.3 and Figure 2.
allocating free blocks of said non-volatile storage means for said block of inode file data and one or more modified blocks of blockmap file data:	The modified blocks are written to disk – see section 4.2. See also section 4.3 and Figure 2.
updating said inode referencing said one or more blocks of blockmap file data to reference said one or more free blocks of said non-volatile storage means allocated to said one or more modified blocks of blockmap file data;	See sections 4.2 and 4.3 and Figure 2.
copying said updated inode to said block of inode file data;	See sections 4.2 and 4.3 and Figure 2.
updating said one or more blocks of blockmap file data;	See sections 4.2 and 4,3 and Figure 2.
writing said updated one or more blocks of blockmap file data and said block of inode file data to said allocated free blocks of said non-volatile storage means.	See sections 4.2 and 4.3 and Figure 2.

Claim 3 of the '292 Patent is also anticipated by Rosenblum II as follows:

US Patent No. 5,819,292	Rosenblum II
3. The method of claim 2 wherein said blocks of meta-file data comprise one or more blocks of inode file data and one or more blocks of blockmap file data and wherein said step of copying said modified blocks of meta-data file data to free blocks of said non-volatile storage means further comprises the steps of:	An inode file data is disclosed in section 4.1. See also section 4.2.1.
copying an inode referencing one or more blocks of blockmap file data to a block of inode file data maintained in said memory means;	The inode file data is modified in memory – see section 4.1. see also section 4.2.1.
allocating free blocks of said non-volatile storage means for said block of inode file data and one or more modified blocks of blockmap file data:	The modified blocks are written to disk – see section 4.1. See also section 4.2.1.
updating said inode referencing said one or more blocks of blockmap file data to reference said one or more free blocks of said non-volatile storage means allocated to said one or more modified blocks of blockmap file data;	See sections 4.2 and 4.3 and in particular sections 4.2.1 and section 4.3.1 through 4.3.5.
copying said updated inode to said block of inode file data;	See sections 4.2 and 4.3.
updating said one or more blocks of blockmap file data;	See section 4.4.
writing said updated one or more blocks of blockmap file data and said block of inode file data to said allocated free blocks of said non-volatile storage means.	See section 4.4.

Claim 4

Claim 4 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
4. A method for maintaining a file system comprising blocks of data stored in blocks of a non-volatile storage means at successive consistency points comprising the steps of:	Section 4.3 and Figure 2 of Borg discloses a file stored in a non-volatile means (a disk) blocks of regular file data and blocks of meta-data file referencing the blocks of data of the file system, at a first consistency point (see Figure 2a).
storing a first file system information structure for a first consistency point in said non-volatile storage means, said first file system information structure comprising data	See section 4.3 and Figure 2.

US Patent No. 5,819,292	Borg
describing a layout of said file system at said first consistency point of said file system;	
writing blocks of data of said file system that have been modified from said first consistency point as of the commencement of a second consistency point to free blocks of said non-volatile storage means;	See section 4.3 and Figure 2.
storing in said non-volatile storage means a second file system information structure for said second consistency point, said second file system information structure comprising data describing a layout said file system at said second consistency point of said file system.	See section 4.3 and Figure 2.

Claim 4 of the '292 Patent is also anticipated by Rosenblum II as follows:

US Patent No. 5,819,292	Rosenblum II
4. A method for maintaining a file system comprising blocks of data stored in blocks of a non-volatile storage means at successive consistency points comprising the steps of:	See section 4.4 and in particular section 4.4.1
storing a first file system information structure for a first consistency point in said non-volatile storage means, said first file system information structure comprising data describing a layout of said file system at said first consistency point of said file system;	See sections 4.2, 4.3 and 4.4.
writing blocks of data of said file system that have been modified from said first consistency point as of the commencement of a second consistency point to free blocks of said non-volatile storage means;	See sections 4.2, 4.3 and 4.4.
storing in said non-volatile storage means a second file system information structure for said second consistency point, said second file system information structure comprising data describing a layout said file system at said second consistency point of said file system.	See sections 4.2-4.4.

<u>Claim 8</u>

Claim 8 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
8. A method for creating a plurality of read-only copies of a file system stored in blocks of a non-volatile storage means, said file system comprising meta-data identifying blocks of said non-volatile storage means used by said file system, comprising the steps of:	Section 4.2 and 4.3 and Figure 2 discloses the formation of a plurality of consistency points. As each new consistency point is created, the "old" consistency point become a read-only copy.
storing meta-data for successive states of said file system in said non-volatile storage means;	See section 4.2, 4.3 and Figure 2 wherein meta-data is also stored in the disk.
making a copy of said meta-data at each of a plurality of said states of said file system;	See Figure 2 and sections 4.2 and 4.3.
for each of said copies of said meta-data at a respective state of said file system, marking said blocks of said non-volatile storage means identified in said meta-data as comprising a respective read-only copy of said file system.	See Figure 2 and sections 4.2 and 4.3 wherein the data associated with an 'old" consistency point becomes "read only"

Claim 8 of the '292 Patent is also anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
8. A method for creating a plurality of read-only copies of a file system stored in blocks of a non-volatile storage means, said file system comprising meta-data identifying blocks of said non-volatile storage means used by said file system, comprising the steps of:	Section 4.1 and Figure 5 show a long term checkpoint with a plurality of saves.
storing meta-data for successive states of said file system in said non-volatile storage means;	See section 4.1 and Figure 5.
making a copy of said meta-data at each of a plurality of said states of said file system;	See section 4.1 and Figure 5.
for each of said copies of said meta-data at a respective state of said file system, marking said blocks of said non-volatile storage means identified in said meta-data as comprising a respective read-only copy of said file system.	See section 4.1 and Figure 5

Claim 8 of the '292 Patent is also anticipated by Hecht as follows:

US Patent No. 5,819,292	Hecht
8. A method for creating a plurality of read-only copies of a file system stored in blocks of a non-volatile storage means, said file system comprising meta-data identifying blocks of said non-volatile storage means used by said	See section 2.3 in which a checkpoint or a "snapshot" with read-only access is periodically taken and transferred to disk.

US Patent No. 5,819,292	Hecht
file system, comprising the steps of:	
storing meta-data for successive states of said file system in said non-volatile storage means;	See section 2.3.
making a copy of said meta-data at each of a plurality of said states of said file system;	See section 2.3.
for each of said copies of said meta-data at a respective state of said file system, marking said blocks of said non-volatile storage means identified in said meta-data as comprising a respective read-only copy of said file system.	See section 2.3.

Claim 9

Claim 9 of the '292 Patent is anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
9. The method of claim 8 wherein said step of marking said blocks comprising a respective read-only copy of said file system comprises placing an appropriate entry in a means for recording multiple usage bits per block of said non-volatile storage means.	Section 3.1 discloses the existence of a STATUS bit, associated with the data.

Claim 10

Claim 10 of the '292 Patent is anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
10. The method of claim 9 wherein said means for recording multiple usage bits per block of said non-volatile storage means comprises a blockmap comprising multiple bit entries for each block.	Lorie discloses a STATUS bit as discussed for claim 9. Figure 3 shows a plurality of STATUS bits associated with a plurality of slots, which is collectively a block. Thus, multiple bits are associated with each block.

Claim 11

Claim 11 of the '292 Patent is obvious in view of Hecht as follows:

US Patent No. 5,819,292	Hecht
11. The method of claim 8 wherein said meta-	Hecht discloses in section 4.2 (page 509) the

US Patent No. 5,819,292	Hecht
data comprises pointers to a hierarchical tree of blocks comprising said file system.	use of pointers. The use of pointers to a hierarchical tree of blocks for the meta-data would have been obvious to one of ordinary skill in the art.

Claim 12

Claim 12 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
12. The method of claim 8 wherein said meta-data comprises structures representing files of said file system.	Borg shows in sections 4.2, 4.3 and Figure 2 meta-data as files.

Claim 12 of the '292 Patent is also anticipated by Rosenblum II as follows:

US Patent No. 5,819,292	Rosenblum II
12. The method of claim 8 wherein said meta-data comprises structures representing files of said file system.	Sections 4.2-4.4 discloses meta-data as files.

Claim 13

Claim 13 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
13. The method of claim 12 wherein said structures representing files of said file system comprise inodes.	Borg is applied to claim 12 and section 4.2 discloses the use of inodes.

Claim 13 of the '292 Patent is also anticipated by Rosenblum II as follows:

US Patent No. 5,819,292	Rosenblum II
13. The method of claim 12 wherein said structures representing files of said file system comprise inodes.	Rosenblum II is applied to claim 12 and section 4.1 discloses the use of inodes.

Claim 14

Claim 14 of the '292 Patent is anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
14. The method of claim 8 further comprising the step of:	
preventing overwriting of said blocks marked as belonging to a read-only copy of said file system.	Lorie discloses the use of a STATUS bit (see section 3). The STATUS bit indicates whether the block can be rewritten or not.

Claim 15

Claim 15 of the '292 Patent is anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
15. The method of claim 8 comprising the step of unmarking said blocks marked as belonging to a read only copy of said file system when said read only copy of said file system is no longer needed.	Lorie discloses the use of a STATUS bit (see section 3). The STATUS bit indicates whether the block can be rewritten or not.

Claim 16

Claim 16 of the '292 Patent is anticipated by Lorie as follows:

US Patent No. 5,819,292	Lorie
16. The method of claim 8 wherein a plurality of said blocks marked as belonging to a read-only copy of said file system comprise data ancillary to said file system, said method further including the steps of:	See sections 3.1 and 3.2.
allowing said ancillary data to be overwritten; and	See sections 3.1 and 3.2 (and Figures 3 and 4) wherein certain segments are overwritten
otherwise preventing overwriting of said blocks marked as comprising a read only copy of said file system	See sections 3.1 and 3.2 and Figures 3 and 4.

Claim 17

Claim 17 of the '292 Patent is obvious in view of Lorie as follows:

US Patent No. 5,819,292	Lorie
17. The method of claim 16 wherein said ancillary data comprises access time data.	Whether the ancillary data is access time data would be a matter of design choice.

Claim 18

Claim 18 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
18. The method of claim 8 wherein said meta-data comprises a root structure referencing structures representing files of said file system, and wherein said copies of said meta-data comprise copies of said root structure.	See Sections 4.2 and 4.3 and Figure 2.

Claim 18 of the '292 Patent is also anticipated by Rosenblum II as follows:

US Patent No. 5,819,292	Rosenblum II
18. The method of claim 8 wherein said meta-data comprises a root structure referencing structures representing files of said file system, and wherein said copies of said meta-data comprise copies of said root structure.	See Sections 4.1-4.4.

Claim 19

Claim 19 of the '292 Patent is anticipated by Borg as follows:

US Patent No. 5,819,292	Borg
19. The method of claim 18 wherein said root structure comprises a root inode.	Borg is applied to claim 18 and section 4.2 discloses inode.

Claim 19 of the '292 Patent is also anticipated by Rosenblum II as follows:

US Patent No. 5,819,292	Rosenblum II
19. The method of claim 18 wherein said root structure comprises a root inode.	Rosenblum II is applied to claim 18 and section 4.1-4.4 discloses inode.

Claim 20

Claim 20 of the '292 Patent is anticipated by either of Borg, Lorie or Hecht as follows:

US Patent No. 5,819,292	Borg, Lorie or Hecht
20. The method of claim 8 further comprising the step of using one or more of said read-only copies of said file system to back-up said blocks comprising one or more consistency points of said file system.	Borg – see sections 4.2 and 4.3 and Figure 2. Lorie – see section 4.1. Hecht – see section 2.3.

III. STATEMENT POINTING OUT SUBSTANTIAL NEW QUESTION OF PATENTABILITY

Since claims 1-4 and 8-20 of the '292 Patent are not patentable over the prior art references cited above for the reasons set forth above, a substantial new question of patentability is raised for each claim. Further, these prior art references cited above are material to the subject matter of the '292 Patent. In particular, these prior art references provide teachings not provided during the prosecution of the '292 Patent. Therefore, a substantial new question of patentability has been raised, and reexamination is respectfully requested.

CONCLUSION

Based on the above remarks, it is respectfully submitted that a substantial new question of patentability has been raised with respect to Claims 1-4 and 8-20 of the '292 Patent. Therefore, reexamination of Claims 1-4 and 8-20 is respectfully requested.

Any fee due for this reexamination may be charged to Deposit Account No. 07-1896.

Respectfully submitted,

DLA PIPER US LLP

Date: October 25, 2007

By: Ronald L. Yin
Ronald L. Yin
Reg. No. 27,607

Attorneys for Applicant(s)

Ronald L. Yin
DLA Piper US LLP
2000 University Avenue
East Palo Alto, CA 94303-2248
650-833-2437 (Direct)
650-833-2000 (Main)
650-833-2001 (Facsimile)
ronald.yin@dlapiper.com

05-14-01

A

PTO/SB/05 (03-01)
Approved for use through 10/31/2002 OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Please type a plus sign (+) inside this box → [+]

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.	103.1068.01
First Inventor	Steven R. KLEIMAN
Title	FILE SYSTEM IMAGE TRANSFER
Express Mail Label No.	EL 734 816 208 US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

1. [ ] Fee Transmittal Form (e.g., PTO/SB/17) *(Submit an original and a duplicate for fee processing)*
2. [ ] Applicant claims small entity status. See 37 CFR 1.27.
3. [X] Specification [*Total Pages* 63] *(preferred arrangement set forth below)*
   - Descriptive title of the invention
   - Cross Reference to Related Applications
   - Statement Regarding Fed sponsored R & D
   - Reference to sequence listing, a table, or a computer program listing appendix
   - Background of the Invention
   - Brief Summary of the Invention
   - Brief Description of the Drawings *(if filed)*
   - Detailed Description
   - Claim(s)
   - Abstract of the Disclosure
4. [X] Drawing(s) *(35 U.S.C. 113)* [*Total Sheets* 4]
5. Oath or Declaration [*Total Pages*]
   a. [ ] Newly executed (original or copy)
   b. [ ] Copy from a prior application (37 CFR 1.63 (d)) *(for continuation/divisional with Box 18 completed)*
      i. [ ] DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
6. [X] Application Data Sheet. See 37 CFR 1.76
7. [ ] CD-ROM or CD-R in duplicate, large table or Computer Program *(Appendix)*
8. Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. [ ] Computer Readable Form (CRF)
   b. Specification Sequence Listing on:
      i. [ ] CD-ROM or CD-R (2 copies); or
      ii. [ ] paper
   c. [ ] Statements verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

9. [ ] Assignment Papers (cover sheet & document(s))
10. [ ] 37 CFR 3.73(b) Statement *(when there is an assignee)* [ ] Power of Attorney
11. [ ] English Translation Document *(if applicable)*
12. [ ] Information Disclosure Statement (IDS)/PTO-1449 [ ] Copies of IDS Citations
13. [ ] Preliminary Amendment
14. [X] Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
15. [ ] Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
16. [ ] Nonpublication Request under 35 U.S.C. 122 (b)(2)(B)(i). Applicant must attach form PTO/SB/35 or its equivalent.
17. [x] Other: Certificate of Mailing, Preliminary Filing

18. If a CONTINUING APPLICATION, *check appropriate box, and supply the requisite information below and in a preliminary amendment, or in an Application Data Sheet under 37 CFR 1.76:*

[ ] Continuation [ ] Divisional [x] Continuation-in-part (CIP) of prior application No.: 09 / 127,497 ; 09 / 153,094

Prior application information Examiner Uyen, Le & Wassum, L *Group Art Unit* 2171 & 2177

For CONTINUATION OR DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 5b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

19. CORRESPONDENCE ADDRESS

[X] *Customer Number or Bar Code Label* 22883 PATENT TRADEMARK OFFICE or [ ] *Correspondence address below*

Name					
Address					
City		*State*		*Zip Code*	
Country		*Telephone*		*Fax*	

Name (Print/Type)	Steven A. Swernofsky	*Registration No. (Attorney/Agent)*	33,040
Signature	S A Swernofsky	*Date*	May 10, 2001

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

22883
PATENT TRADEMARK OFFICE

103.1068.01

Certificate of Mailing (37 C.F.R. § 1.10)

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Services on the date shown below as "Express Mail" (Post Office to Addressee) in an envelope addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231.

Mailing Label No. **EL 734 816 208 US**

Date of Deposit: May 10, 2001

Arlette Malhas
Printed Name

Arlette Malhas
Signature

Documents enclosed:

- Utility Patent Application Transmittal Form;
- Application Data Sheet;
- Certificate of Express Mail Mailing;
- Preliminary Filing, **6** pages;
- Specification, **50** pgs.;
- Claims, **12** pgs.;
- Abstract, **1** pg.;
- Drawings, **4** pgs.; and
- Return postcard

Application Data Sheet (Multiple Inventors with Representation)

Inventor Information

Inventor One Given Name::	Steven R.
Family Name::	Kleiman
Postal Address Line One::	157 El Monte Road
City::	Los Altos
State or Province::	CA
Postal or Zip Code::	94022
Citizenship Country::	US
Inventor Two Given Name::	David
Family Name::	Hitz
Postal Address Line One::	137 Sylvian Way
City::	Los Altos
State or Province::	CA
Postal or Zip Code::	94022
Citizenship Country::	US
Inventor Three Given Name::	Guy
Family Name::	Harris
Postal Address Line One::	46 Starlite Court
City::	Mountain View
State or Province::	CA
Postal or Zip Code::	94043-1937
Citizenship Country::	US
Inventor Four Given Name::	Sean W.
Family Name::	O'Malley
Postal Address Line One::	4820 N. Camino Luz
City::	Tucson
State or Province::	AZ
Postal or Zip Code::	95718
Citizenship Country::	US
Inventor Five Given Name::	Michael
Family Name::	Malcolm
Postal Address Line One::	199 Yerba Buena Avenue
City::	Los Altos
State or Province::	CA
Postal or Zip Code::	94022

Inventor Six Given Name::	James
Family Name::	Lau
Postal Address Line One::	12894 La Cresta Drive
City::	Los Altos Hills
State or Province::	CA
Postal or Zip Code::	94022
Citizenship Country::	US
Inventor Seven Given Name::	Byron
Family Name::	Rakitzis
Postal Address Line One::	149 Occidental Avenue
City::	Burlingame
State or Province::	CA
Postal or Zip Code::	94010
Citizenship Country::	US

Correspondence Information

Correspondence Customer Number::	22883
Telephone One::	(650) 947-0700
Fax::	(650) 947-8438
Electronic Mail::	arlette@swernofsky.com

Application Information

Title Line One::	File System
Title Line Two::	Image Transfer
Total Drawing Sheets::	4
Formal Drawings?::	No
Application Type::	Utility
Docket Number::	103.1068.01

Representative Information

Representative Customer Number::	22883

Continuity Information

This Application is a::	Continuation-in-Part of
>Application One::	09/127,497
Filing Date::	July 31, 1998

This Application is a::	Continuation-in-Part of
>Application Two::	09/153,094
Filing Date::	September 14, 1998
which is a::	Continuation of
>>Application Three::	09/108,022
Filing Date::	June 30, 1998
Patent Number::	5,963,962
which is a::	Continuation of
>>>Application Four::	08/454,921
Filing Date::	May 31, 1995
Patent Number::	5,819,292
which is a::	Continuation of
>>>>Application Five::	08/071,643
Filing Date::	June 3, 1993

22883
PATENT TRADEMARK OFFICE

103.1068.01

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

R.
STEVEN KLEIMAN et al.

Serial No.: N/Y/A (CIP of App. Nos. 09/127,497 & 09/153,094)

Filed: Herewith

For: File System Image Transfer

Art Unit: N/Y/A (parents: 2171 & 2177)

Examiner: N/Y/A (parents: UYEN LE & LUKE WASSUM)

09/544,837 (side stamp: 09544837.10015)

Honorable Assistant Commissioner
for Patents
Washington, D.C. 20231

PRELIMINARY FILING

Dear Sir:

This is a preliminary filing for a continuation-in-part of application no. 09/127,497, filed July 31, 1998 and also of application no. 09/153,094, filed Sept. 14, 1998 (now allowed).

103.1068.01

REMARKS:

This paper is being filed with the above-identified CIP application so as to explain the procedural status of the application for the Examiner's benefit and for the record.

Subject Matter of CIP Application

The disclosure of this CIP application is substantially identical to that of parent application no. 09/127,497. Applicants note that this parent application incorporated the disclosure of a parent of parent application no. 09/153,094 by reference. Thus, all mater disclosed in this CIP application is believed to have been disclosed by parent application no. 09/127,497, either directly or by incorporation of a parent of application no. 09/153,094.

The CIP application has been modified to correct formal matters raised in the prosecution of parent application no. 09/127,497, to recite claims rejected in the parent applications, to conform those claims to amendments made in the parents, and to renumber the claims. For the Examiner's reference, the following table shows the correlation between claims of this CIP application and claims in the parent applications.

Claim in CIP Application	Claim in Parent Application
1	1 from app. no. 09/127,497
2	2 from app. no. 09/127,497
3	3 from app. no. 09/127,497
4	4 from app. no. 09/127,497
5	5 from app. no. 09/127,497
6	6 from app. no. 09/127,497
7	7 from app. no. 09/127,497

103.1068.01

Claim in CIP Application	Claim in Parent Application
8	8 from app. no. 09/127,497
9	9 from app. no. 09/127,497
10	13 from app. no. 09/127,497
11	15 from app. no. 09/127,497
12	16 from app. no. 09/127,497
13	17 from app. no. 09/127,497
14	18 from app. no. 09/127,497
15	19 from app. no. 09/127,497
16	23 from app. no. 09/127,497
17	24 from app. no. 09/127,497
18	25 from app. no. 09/127,497
19	26 from app. no. 09/127,497
20	27 from app. no. 09/127,497
21	28 from app. no. 09/127,497
22	34 from app. no. 09/127,497
23	35 from app. no. 09/127,497
24	36 from app. no. 09/127,497
25	43 from app. no. 09/127,497
26	51 from app. no. 09/127,497
27	58 from app. no. 09/127,497
28	59 from app. no. 09/127,497
29	60 from app. no. 09/127,497
30	61 from app. no. 09/127,497
31	62 from app. no. 09/127,497
32	63 from app. no. 09/127,497
33	67 from app. no. 09/127,497
34	68 from app. no. 09/127,497
35	70 from app. no. 09/127,497
36	72 from app. no. 09/127,497
37	73 from app. no. 09/127,497
38	74 from app. no. 09/127,497
39	75 from app. no. 09/127,497
40	76 from app. no. 09/127,497
41	78 from app. no. 09/127,497
42	79 from app. no. 09/127,497
43	80 from app. no. 09/127,497
44	81 from app. no. 09/127,497
45	83 from app. no. 09/127,497

103.1068.01

Claim in CIP Application	Claim in Parent Application
46	84 from app. no. 09/127,497
47	85 from app. no. 09/127,497
48	86 from app. no. 09/127,497
49	88 from app. no. 09/127,497
50	91 from app. no. 09/127,497
51	93 from app. no. 09/127,497
52	3 from app. no. 09/153,094
53	3 from app. no. 09/153,094 (modified to remove "means" terminology)

Rejection of Claims from Parent application no. 09/127,497

Claims 1 to 51 correspond to claims rejected in parent application no. 09/127,497 under 35 U.S.C. § 102(e) or § 103(a) over U.S. Patent No. 5,819,292 (Hitz). Hitz is a parent of a parent of this CIP application, namely application no. 09/153,094. This CIP application therefore claims the same priority as is claimed by Hitz for any and all claims that recite subject matter disclosed by Hitz. Accordingly, a rejection over Hitz is not permissible.

Rejection of Claims from Parent application no. 09/153,094

Claim 52 corresponds to claim 3 rejected in parent application no. 09/153,094 under the judicially created doctrine of obviousness-type double patenting over claim 1 of U.S. Patent No. 5,819,292 (Hitz). Claim 53 is a version of claim 52 rewritten so as to remove "means for" terminology. Applicants intend to file a terminal disclaimer if and when a double patenting rejection over Hitz is the only outstanding issue in this case.

Inventorship

In an earlier response filed in parent application no. 09/127,497, Applicants declined to concede that the rejection of claims in that application over Hitz was proper. Applicants maintain that at least some of the claims in this CIP application that correspond to rejected claims in parent application no. 09/127,497 are not disclosed or suggested by Hitz. For example, Hitz does not use the term "shadow snapshot," which is recited by some of these claims. Accordingly, the inventors for this CIP application also include the inventors for parent application no. 09/127,497.

However, upon further consideration, Applicants concede that at least claim 34 from application no. 09/127,497, which is now claim 22, was fully disclosed by Hitz. Accordingly, this claim finds priority in the parent of Hitz, namely application no. 08/071,643, and the inventors for this CIP application include the inventors for this parent application. Furthermore, the inventors for Hitz are included because claims 52 and 53 are taken from a continuation of Hitz.

In view of the foregoing, inventorship for this CIP application is believed to be correct.

Closing

The entire application is believed to be in condition for allowance. Early passage to issue is respectfully requested at the Examiner's earliest convenience.

Applicants' undersigned attorney can be reached at (614) 486-3585. All correspondence should continue to be directed to the address indicated below.

Respectfully submitted,

Dane C. Butzer

Dated: April 30, 2001

Dane C. Butzer
Reg. No. 43,521

The Swernofsky Law Group
P.O. Box 390013
Mountain View, CA 94039-0013
(650) 947-0700

09854167.051001

This application is submitted in the name of the following inventors:

Inventor	Citizenship	Residence City and State
KLEIMAN, Steven R	United States	Los Altos, California
HITZ, David	United States	Los Altos, California
HARRIS, Guy	United States	Mountain View, California
O'MALLEY, Sean	United States	Tuscon, Arizona
MALCOM, Michael	United States	Los Altos, California
LAU, James	United States	Los Altos Hills, California
RAKITZIS, Byron	United States	Burlingame, California

The assignee is Network Appliance, Inc., a corporation having an office at 495 Java Drive, Sunnyvale, CA 94089.

Cross Reference to Related Applications

This is a continuation-in-part of Application No. 09/127,497, filed July 31, 1998. This is also a continuation-in-part of Application No. 09/153,094, filed Sept. 14, 1998 (now allowed), which is a continuation of Application No. 09/108,022, filed June 30, 1998 (now U.S. Patent No. 5,963,962), which is a continuation of Application No. 08/454,921, filed May 31, 1995 (now U.S. Patent No. 5,819,292), which is a continuation of Application No. 08/071,643, filed June 3, 1993 (now abandoned).

103.1068.01

Title of the Invention

File System Image Transfer

Background of the Invention

1. *Field of the Invention*

The invention relates to storage systems.

2. *Related Art*

In computer file systems for storing and retrieving information, it is sometimes advantageous to duplicate all or part of the file system. For example, one purpose for duplicating a file system is to maintain a backup copy of the file system to protect against lost information. Another purpose for duplicating a file system is to provide replicas of the data in that file system available at multiple servers, to be able to share load incurred in accessing that data.

One problem in the known art is that known techniques for duplicating data in a file system either are relatively awkward and slow (such as duplication to tape), or are relatively expensive (such as duplication to an additional set of disk drives). For

example, known techniques for duplication to tape rely on logical operations of the file system and the logical format of the file system. Being relatively cumbersome and slow discourages frequent use, resulting in backup copies that are relatively stale. When data is lost, the most recent backup copy might then be a day old, or several days old, severely reducing the value of the backup copy.

Similarly, known techniques for duplication to an additional set of disk drives rely on the physical format of the file system as stored on the original set of disk drives. These known techniques use an additional set of disk drives for duplication of the entire file system. Being relatively expensive discourages use, particularly for large file systems. Also, relying on the physical format of the file system complicates operations for restoring backup data and for performing incremental backup.

Accordingly, it would be desirable to provide a method and system for duplicating all or part of a file system, which can operate with any type of storage medium without either relative complexity or expense, and which can provide all the known functions for data backup and restore. This advantage is achieved in an embodiment of the invention in which consistent copies of the file system are maintained, so those consistent snapshots can be transferred at a storage block level using the file server's own block level operations.

Summary of the Invention

The invention provides a method and system for duplicating all or part of a file system while maintaining consistent copies of the file system. The file server maintains a set of snapshots, each indicating a set of storage blocks making up a consistent copy of the file system as it was at a known time. Each snapshot can be used for a purpose other than maintaining the coherency of the file system, such as duplicating or transferring a backup copy of the file system to a destination storage medium. In a preferred embodiment, the snapshots can be manipulated to identify sets of storage blocks in the file system for incremental backup or copying, or to provide a file system backup that is both complete and relatively inexpensive.

Brief Description of the Drawings

Figure 1 shows a block diagram of a first system for file system image transfer.

Figure 2 shows a block diagram of a set of snapshots in a system for file system image transfer.

Figure 3 shows a process flow diagram of a method for file system image transfer.

Detailed Description of the Preferred Embodiment

In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using one or more general purpose processors (or special purpose processors adapted to the particular process steps and data structures) operating under program control, and that implementation of the preferred process steps and data structures described herein using such equipment would not require undue experimentation or further invention.

Inventions described herein can be used in conjunction with inventions described in the following applications:

- Application Serial No. 08/471,218, filed June 5, 1995, in the name of inventors David Hitz et al., titled "A Method for Providing Parity in a Raid Sub-System Using Non-Volatile Memory", now U.S. Patent No. 5,948,110;

- Application Serial No. 08/454,921, filed May 31, 1995, in the name of inventors David Hitz et al., titled "Write Anywhere File-System Layout", now U.S. Patent No. 5,819,292;

- Application Serial No. 08/464,591, filed May 31, 1995, in the name of inventors David Hitz et al., titled "Method for Allocating Files in a File System Integrated with a Raid Disk Sub-System", now U.S. Patent No. 6,038,570.

Each of these applications is hereby incorporated by reference as if fully set forth herein. They are collectively referred to as the "WAFL Disclosures."

File Servers and File System Image Transfer

Figure 1 shows a block diagram of a system for file system image transfer.

A system 100 for file system image transfer includes a file server 110 and a destination file system 120.

The file server 110 includes a processor 111, a set of program and data memory 112, and mass storage 113, and preferably is a file server like one described in the WAFL Disclosures. In a preferred embodiment, the mass storage 113 includes a RAID storage subsystem and stores data for file system 114.

The destination file system 120 includes mass storage, such as a flash memory, a magnetic or optical disk drive, a tape drive, or other storage device. In a preferred embodiment, the destination file system 120 includes a RAID storage

subsystem. The destination file system 120 can be coupled directly or indirectly to the file server 110 using a communication path 130.

In a first preferred embodiment, the destination file system 120 is coupled to the file server 110 and controlled by the processor 111 similarly to the mass storage 113. In this first preferred embodiment, the communication path 130 includes an internal bus for the file server 110, such as an I/O bus, a mezzanine bus, or other system bus.

In a second preferred embodiment, the destination file system 120 is included in a second file server 140. The second file server 140, similar to the first file server 110, includes a processor, a set of program and data memory, and mass storage that serves as the destination file system 120 with regard to the first file server 110. The second file server preferably is a file server like one described in the WAFL Disclosures. In this second preferred embodiment, the communication path 130 includes a network path between the first file server 110 and the second file server 140, such as a direct communication link, a LAN (local area network), a WAN (wide area network), a NUMA network, or another interconnect.

In a third preferred embodiment, the communication path 130 includes an intermediate storage medium, such as a tape, and the destination file system 120 can be either the first file server 110 itself or a second file server 140. As shown below, when the file server 110 selects a set of storage blocks for transfer to the destination file system

120, that set of storage blocks can be transferred by storing them onto the intermediate storage medium. At a later time, retrieving that set of storage blocks from the intermediate storage medium completes the transfer.

It is an aspect of the invention that there are no particular restrictions on the communication path 130. For example, a first part of the communication path 130 can include a relatively high-speed transfer link, while a second part of the communication path 130 can include an intermediate storage medium.

It is a further aspect of the invention that the destination file system 120 can be included in the first file server 110, in a second file server 140, or distributed among a plurality of file servers. Transfer of storage blocks from the first file server 110 to the destination file system 120 is thus completely general, and includes the possibility of a wide variety of different file system operations:

- Storage blocks from the first file server 110 can be dumped to an intermediate storage medium, such as a tape or a second disk drive, retained for a period of time, and then restored to the first file server 110. Thus, the first file server 110 can itself be the destination file system.

- Storage blocks from the first file server 110 can be transferred to a second file server 140, and used at that second file server 140. Thus, the storage blocks can be copied en masse from the first file server 110 to the second file server 140.

- Storage blocks from the first file server 110 can be distributed using a plurality of different communication paths 130, so that some of the storage blocks are immediately accessible while others are recorded in a relatively slow intermediate storage medium, such as tape.

- Storage blocks from the first file server 110 can be selected from a complete file system, transferred using the communication path 130, and then processed to form a complete file system at the destination file system 120.

In alternative embodiments described herein, the second file server 140 can have a second destination file system. That second destination file system can be included within the second file server 140, or can be included within a third file server similar to the first file server 110 or the second file server 140.

More generally, each n^{th} file server can have a destination file system, either included within the n^{th} file server, or included within an $n+1^{st}$ file server. The set of file servers can thus form a directed graph, preferably a tree with the first file server 110 as the root of that tree.

File System Storage Blocks

As described in the WAFL Disclosures, a file system 114 on the file server 110 (and in general, on the n^{th} file server), includes a set of storage blocks 115, each of which is stored either in the memory 112 or on the mass storage 113. The file system 114 includes a current block map, which records which storage blocks 115 are part of the file system 114 and which storage blocks 115 are free.

As described in the WAFL Disclosures, the file system on the mass storage 113 is at all times consistent. Thus, the storage blocks 115 included in the file system at all times comprise a consistent file system 114.

As used herein, the term "consistent," referring to a file system (or to storage blocks in a file system), means a set of storage blocks for that file system that includes all blocks required for the data and file structure of that file system. Thus, a consistent file system stands on its own and can be used to identify a state of the file system at some point in time that is both complete and self-consistent.

As described in the WAFL Disclosures, when changes to the file system 114 are committed to the mass storage 113, the block map is altered to show those storage blocks 115 that are part of the committed file system 114. In a preferred embodiment, the file server 110 updates the file system frequently, such as about once each 10 seconds.

Snapshots

Figure 2 shows a block diagram of a set of snapshots in a system for file system image transfer.

As used herein, a "snapshot" is a set of storage blocks, the member storage blocks forming a consistent file system, disposed using a data structure that allows for efficient set management. The efficient set management can include time efficiency for set operations (such as logical sum, logical difference, membership, add member, remove member). For example, the time efficiency can include O(n) time or less for n storage blocks. The efficient set management can also include space efficiency for enumerating the set (such as association with physical location on mass storage or inverting the membership function). The space efficiency can mean about 4 bytes or less per 4K storage block of disk space, a ratio about 1000:1 better than duplicating the storage space.

As described herein, the data structure for the snapshot is stored in the file system so there is no need to traverse the file system tree to recover it. In a preferred embodiment, each snapshot is stored as a file system object, such as a blockmap. The blockmap includes a bit plane having one bit for each storage block, other than bits used to identify if the storage block is in the active file system.

Moreover, when the file system is backed-up, restored, or otherwise copied or transferred, the blockmap within the file system is as part of the same operation itself also backed-up, restored, or otherwise copied or transferred. Thus, operations on the file system inherently include preserving snapshots.

Any particular snapshot can be transferred by any communication technique, including

- o transfer using storage in an intermediate storage medium (such as nonvolatile memory, tape, disk in the same file system, disk in a different file system, or disk distributed over several file systems);

- o transfer using one or more network messages,

- o transfer using communication within a single file server or set of file servers (such as for storage to disk in the same file system, to disk in a different file system, or to disk distributed over several file systems).

A collection 200 of snapshots 210 includes one bit plane for each snapshot 210. Each bit plane indicates a set of selected storage blocks 115. In the figure, each column indicates one bit plane (that is, one snapshot 210), and each row indicates one storage block 115 (that is, the history of that storage block 115 being included in or

excluded from successive snapshots 210). At the intersection of each column and each row there is a bit 211 indicating whether that particular storage block 115 is included in that particular snapshot 210.

Each snapshot 210 comprises a collection of selected storage blocks 115 from the file system 114 that formed all or part of the (consistent) file system 114 at some point in time. A snapshot 210 can be created based on the block map at any time by copying the bits from the block map indicating which storage blocks 115 are part of the file system 114 into the corresponding bits 211 for the snapshot 210.

Differences between the snapshots 210 and the (active) file system 114 include the following:

- The file system 114 is a consistent file system 114 that is being used and perhaps modified, while the snapshots 210 represent copies of the file system 114 that are read-only.

- The file system 114 is updated frequently, while the snapshots 210 represent copies of the file system 114 that are from the relatively distant past.

- There is only one active file system 114, while there can be (and typically are) multiple snapshots 210.

At selected times, the file server 110 creates a new bit plane, based on the block map, to create a new snapshot 210. As described herein, snapshots 210 are used for backup and mirroring of the file system 114, so in preferred embodiments, new snapshots 210 are created at periodic times, such as once per hour, day, week, month, or as otherwise directed by an operator of the file server 110.

Storage Images and Image Streams

As used herein a "storage image" includes an indicator of a set of storage blocks selected in response to one or more snapshots. The technique for selection can include logical operations on sets (such as pairs) of snapshots. In a preferred embodiment, these logical operations can include logical sum and logical difference.

As used herein, an "image stream" includes a sequence of storage blocks from a storage image. A set of associated block locations for those storage blocks from the storage image can be identified in the image stream either explicitly or implicitly. For a first example, the set of associated block locations can be identified explicitly by including volume block numbers within the image stream. For a second example, the set of associated block locations can be identified implicitly by the order in which the storage blocks from the storage image are positioned or transferred within the image stream.

The sequence of storage blocks within the image stream can be optimized for a file system operation. For example, the sequence of storage blocks within the image stream can be optimized for a backup or restore file system operation.

In a preferred embodiment, the sequence of storage blocks is optimized so that copying of an image stream and transfer of that image stream from one file server to another is optimized. In particular, the sequence of storage blocks is selected so that storage blocks identified in the image stream can be, as much as possible, copied in parallel from a plurality of disks in a RAID file storage system, so as to maximize the transfer bandwidth from the first file server.

A storage image 220 comprises a set of storage blocks 115 to be copied from the file system 114 to the destination file system 120.

The storage blocks 115 in the storage image 220 are selected so that when copied, they can be combined to form a new consistent file system 114 on the destination file system 120. In various preferred embodiments, the storage image 220 that is copied can be combined with storage blocks 115 from other storage images 220 (which were transferred at earlier times).

As shown herein, the file server 110 creates each storage image 220 in response to one or more snapshots 210

An image stream 230 comprises a sequence of storage blocks 115 from a storage image 220. When the storage image 220 is copied from the file system 114, the storage blocks 115 are ordered into the image stream 230 and tagged with block location information. When the image stream 230 is received at the destination file system 120, the storage blocks 115 in the image stream 230 are copied onto the destination file system 120 in response to the block location information.

Image Addition and Subtraction

The system 100 manipulates the bits 211 in a selected set of storage images 220 to select sets of storage blocks 115, and thus form a new storage image 220.

For example, the following different types of manipulation are possible:

- The system 100 can form a logical sum of two storage images 220 A + B by forming a set of bits 211 each of which is the logical OR (A v B) of the corresponding bits 211 in the two storage images 220. The logical sum of two storage images 220 A + B is the union of those two storage images 220.

- The system 100 can form a logical difference of two storage images 220 A – B by forming a set of bits 211 each of which is logical "1" only if the corresponding bit

211 A is logical "1" and the corresponding bit 211 B is logical "0" in the two storage images 220.

The logical sum of two storage images 220 A + B comprises a storage image 220 that includes storage blocks 115 in either of the two original storage images 220. Using the logical sum, the system 100 can determine not just a single past state of the file system 114, but also a history of past states of that file system 114 that were recorded as snapshots 210.

The logical difference of two selected storage images 220 A – B comprises just those storage blocks that are included in the storage image 220 A but not in the storage image 220 B. (To preserve integrity of incremental storage images, the subtrahend storage image 220 B is always a snapshot 210.) A logical difference is useful for determining a storage image 220 having a set of storage blocks forming an incremental image, which can be used in combination with full images.

In alternative embodiments, other and further types of manipulation may also be useful. For example, it may be useful to determine a logical intersection of snapshots 210, so as to determine which storage blocks 115 were not changed between those snapshots 210.

In further alternative embodiments, the system 100 may also use the bits 211 from each snapshot 210 for other purposes, such as to perform other operations on the storage blocks 115 represented by those bits 211.

Incremental Storage Images

As used herein, an "incremental storage image" is a logical difference between a first storage image and a second storage image.

As used herein, in the logical difference A – B, the storage image 220 A is called the "top" storage image 220, and the storage image 220 B is called the "base" storage image 220.

When the base storage image 220 B comprises a full set F of storage blocks 115 in a consistent file system 114, the logical difference A – B includes those incremental changes to the file system 114 between the base storage image 220 B and the top storage image 220 A.

Each incremental storage image 220 has a top storage image 220 and a base storage image 220. Incremental storage images 220 can be chained together when there is a sequence of storage images 220 C_i where a base storage image 220 for each C_i is a top storage image 220 for a next C_{i+1}.

Examples of Incremental Images

For a first example, the system 100 can make a snapshot 210 each day, and form a level-0 storage image 220 in response to the logical sum of daily snapshots 210.

June3.level0 = June3 + June2 + June1

(June3, June2, and June1 are snapshots 220 taken on those respective dates.)

The June3.level0 storage image 220 includes all storage blocks 115 in the daily snapshots 210 June3, June2, and June1. Accordingly, the June3.level0 storage image 220 includes all storage blocks 115 in a consistent file system 114 (as well as possibly other storage blocks 115 that are unnecessary for the consistent file system 114 active at the time of the June3 snapshot 210).

In the first example, the system 100 can form an (incremental) level-1 storage image 220 in response to the logical sum of daily snapshots 210 and the logical difference with a single snapshot 210.

June5.level1 = June5 + June4 – June3

(June5, June4 and June3 are snapshots 220 taken on those respective dates.)

It is not required to subtract the June2 and June1 snapshots 210 when forming the June5.level1 storage image 220. All storage blocks 115 that the June5 snapshot 210 and the June4 snapshot 210 have in common with either the June2 snapshot 210 or the June1 snapshot 210, they will necessarily have in common with the June3 snapshot 210. This is because any storage block 115 that was part of the file system 114 on June2 or June1, and is still part of the file system 114 on June5 or June4, must have also been part of the file system 114 on June3.

In the first example, the system 100 can form an (incremental) level-2 storage image 220 in response to the logical sum of daily snapshots 210 and the logical difference with a single snapshot 210 from the time of the level-1 base storage image 220.

June7.level2 = June7 + June6 – June5

(June7, June6, and June5 are snapshots 210 taken on those respective dates.)

In the first example, the storage images 220 June3.level0, June5.level1, and June7.level2 collectively include all storage blocks 115 needed to construct a full set F of storage blocks 115 in a consistent file system 114.

For a second example, the system 100 can form a different (incremental) level-1 storage image 220 in response to the logical sum of daily snapshots 210 and the

logical difference with a single snapshot 210 from the time of the level-0 storage image 220.

June9.level1 = June9 + June8 – June3

(June9, June8, and June3 are snapshots 210 taken on those respective dates.)

Similar to the first example, the storage images 220 June3.level0 and June9.level1 collectively include all storage blocks 115 needed to construct a full set F of storage blocks 115 in a consistent file system 114. There is no particular requirement that the June9.level1 storage image 220 be related to or used in conjunction with the June7.level2 storage image 220 in any way.

File System Image Transfer Techniques

To perform one of these copying operations, the file server 110 includes operating system or application software for controlling the processor 111, and data paths for transferring data from the mass storage 113 to the communication path 130 to the destination file system 120. However, the selected storage blocks 115 in the image stream 230 are copied from the file system 114 to the corresponding destination file system 120 without logical file system processing by the file system 114 on the first file server 110.

In a preferred embodiment, the system 100 is disposed to perform one of at least four such copying operations:

- Volume Copying. The system 100 can be disposed to create an image stream 230 for copying the file system 114 to the destination file system 120.

The image stream 230 comprises a sequence of storage blocks 115 from a storage image 220. As in nearly all the image transfer techniques described herein, that storage image 220 can represent a full image or an incremental image:

Full image: The storage blocks 115 and the storage image 220 represent a complete and consistent file system 114.

Incremental image: The storage blocks 115 and the storage image 220 represent an incremental set of changes to a consistent file system 114, which when combined with that file system 114 form a new consistent file system 114.

The image stream 230 can be copied from the file server 110 to the destination file system 120 using any communication technique. This could include a direct communication link, a LAN (local area network), a WAN (wide area network), transfer via tape, or a combination thereof. When the image stream 230 is transferred using a network, the storage blocks 115 are encapsulated in messages using a network

communication protocol known to the file server 110 and to the destination file system 120. In some network communication protocols, there can be additional messages between the file server 110 and to the destination file system 120 to ensure the receipt of a complete and correct copy of the image stream 230.

The destination file system 120 receives the image stream 230 and identifies the storage blocks 115 from the mass storage 113 to be recorded on the destination file system 120.

When the storage blocks 115 represent a complete and consistent file system 114, the destination file system 120 records that file system 114 without logical change. The destination file system 120 can make that file system 114 available for read-only access by local processes. In alternative embodiments, the destination file system 120 may make that file system 114 available for access by local processes, without making changes by those local processes available to the file server 110 that was the source of the file system 114.

When the storage blocks 115 represent an incremental set of changes to a consistent file system 114, the destination file system 120 combines those changes with that file system 114 form a new consistent file system 114. The destination file system 120 can make that new file system 114 available for read-only access by local processes.

In embodiments where the destination file system 120 makes the transferred file system 114 available for access by local processes, changes to the file system 114 at the destination file system 120 can be flushed when a subsequent incremental set of changes is received by the destination file system 120.

All aspects of the file system 114 arc included in the image stream 230, including file data, file structure hierarchy, and file attributes. File attributes preferably include NFS attributes, CIFS attributes, and those snapshots 210 already maintained in the file system 114.

Disk Copying. In a first preferred embodiment of volume copying (herein called "disk copying"), the destination file system 120 can include a disk drive or other similar accessible storage device. The system 100 can copy the storage blocks 115 from the mass storage 113 to that accessible storage device, providing a copy of the file system 114 that can be inspected at the current time.

When performing disk copying, the system 100 creates an image stream 230, and copies the selected storage blocks 115 from the mass storage 113 at the file server 110 to corresponding locations on the destination file system 120. Because the mass storage 113 at the file server 110 and the destination file system 120 are both disk drives, copying to corresponding locations should be simple and effective.

It is possible that locations of storage blocks 115 at the mass storage 113 at the file server 110 and at the destination file system 120 do not readily coincide, such as if the mass storage 113 and the destination file system 120 have different sizes or formatting. In those cases, the destination file system 120 can reorder the storage blocks 115 in the image stream 230, similar to the "Tape Backup" embodiment described herein.

Tape Backup. In a second preferred embodiment of volume copying (herein called "tape backup"), the destination file system 120 can include a tape device or other similar long-term storage device. The system 100 can copy storage blocks 115 from the mass storage 113 to that long-term storage device, providing a backup copy of the file system 114 that can be restored at a later time.

When performing tape backup, the system 100 creates an image stream 230, and copies the selected storage blocks 115 from the mass storage 113 at the file server 110 to a sequence of new locations on the destination file system 120. Because the destination file system 120 includes one or more tape drives, the system 100 creates and transmits a table indicating which locations on the mass storage 113 correspond to which other locations on the destination file system 120.

Similar to transfer of an image stream 230 using a network communication protocol, the destination file system 120 can add additional information to the image stream 230 for recording on tape. This additional information can include tape headers

and tape gaps, blocking or clustering of storage blocks 115 for recording on tape, and reformatting of storage blocks 115 for recording on tape.

File Backup. In a third preferred embodiment of volume copying (herein called "file backup"), the image stream 230 can be copied to a new file within a file system 114, either at the file server 110 or at a file system 114 on the destination file system 120.

Similar to tape backup, the destination file system 120 can add additional information to the image stream 230 for recording in an file. This additional information can include file metadata useful for the file system 114 to locate storage blocks 115 within the file.

- Volume Mirroring. The system 100 can be disposed to create image streams 230 for copying the file system 114 to the destination file system 120 coupled to a second file server on a frequent basis, thus providing a mirror copy of the file system 114.

In a preferred embodiment, the mirror copy of the file system 114 can be used for takeover by a second file server 140 from the first file server 110, such as for example if the first file server 110 fails.

When performing volume mirroring, the system 100 first transfers an image stream 230 representing a complete file system 114 from the file server 110 to the destination file system 120. The system 100 then periodically transfers image streams 230 representing incremental changes to that file system 114 from the file server 110 to the destination file system 120. The destination file system 120 is able to reconstruct a most recent form of the consistent file system 114 from the initial full image stream 230 and the sequence of incremental image streams 230.

It is possible to perform volume mirroring using volume copying of a full storage image 230 and a sequence of incremental storage images 230. However, determining the storage blocks 115 to be included in an incremental storage images 230 can take substantial time for a relatively large file system 114, if done by logical subtraction.

As used herein, a "mark-on-allocate storage image" is a subset of a snapshot, the member storage blocks being those that have been added to a snapshot that originally formed a consistent file system.

In a preferred embodiment, rather than using logical subtraction, as described above, at the time the incremental storage images 230 is about to be transferred, the file server 110 maintains a separate "mark-on-allocate" storage image 230. The mark-on-allocate storage image 230 is constructed by setting a bit for each storage block 115,

as it is added to the consistent file system 114. The mark-on-allocate storage image 230 does not need to be stored on the mass storage 113, included in the block map, or otherwise backed-up; it can be reconstructed from other storage images 230 already at the file server 110.

When an incremental storage image 230 is transferred, a first mark-on-allocate storage image 230 is used to determine which storage blocks 115 to include in the storage image 230 for transfer. A second mark-on-allocate storage image 230 is used to record changes to the file system 114 while the transfer is performed. After the transfer is performed, the first and second mark-on-allocate storage images 230 exchange roles.

Full Mirroring. In a first preferred embodiment of volume mirroring (herein called "full mirroring"), the destination file system 120 includes a disk drive or other similar accessible storage device.

Upon the initial transfer of the full storage image 230 from the file server 110, the destination file system 120 creates a copy of the consistent file system 114. Upon the sequential transfer of each incremental storage image 230 from the file server 110, the destination file system 120 updates its copy of the consistent file system 114. The destination file system 120 thus maintains its copy of the file system 114 nearly up to date, and can be inspected at any time.

When performing full mirroring, similar to disk copying, the system 100 creates an image stream 230, and copies the selected storage blocks 115 from the mass storage 113 at the file server 110 to corresponding locations on the destination file system 120.

Incremental Mirroring. In a second preferred embodiment of volume mirroring (herein called "incremental mirroring"), the destination file system 120 can include both (1) a tapc device or other relatively slow storage device, and (2) a disk drive or other relatively fast storage device.

As used herein, an "incremental mirror" of a first file system is a base storage image from the first file system, and at least one incremental storage image from the first file system, on two storage media of substantially different types. Thus, a complete copy of the first file system can be reconstructed from the two or more objects.

Upon the initial transfer of the full storage image 230 from the file server 110, the destination file system 120 copies a complete set of storage blocks 115 from the mass storage 113 to that relatively slow storage device. Upon the sequential transfer of each incremental storage image 230 from the file server 110, the destination file system 120 copies incremental sets of storage blocks 115 from the mass storage 113 to the relatively fast storage device. Thus, the full set of storage blocks 115 plus the

incremental sets of storage blocks 115 collectively represent an up-to-date file system 114 but do not require an entire duplicate disk drive.

When performing incremental mirroring, for the base storage image 230, the system 100 creates an image stream 230, and copies the selected storage blocks 115 from the mass storage 113 at the file server 110 to a set of new locations on the relatively slow storage device. The system 100 writes the image stream 230, including storage block location information, to the destination file system 120. In a preferred embodiment, the system 100 uses a tape as an intermediate destination storage medium, so that the base storage image 230 can be stored for a substantial period of time without having to occupy disk space.

For each incremental storage image 230, the system 100 creates a new image stream 230, and copies the selected storage blocks 115 from the mass storage 113 at the file server 110 to a set of new locations on the accessible storage device. Incremental storage images 230 are created continuously and automatically at periodic times that are relatively close together.

The incremental storage images 230 are received at the destination file system 120, which unpacks them and records the copied storage blocks 115 in an incremental mirror data structure. As each new incremental storage image 230 is copied, copied storage blocks 115 overwrite the equivalent storage blocks 115 from earlier

incremental storage images 230. In a preferred embodiment, the incremental mirror data structure includes a sparse file structure including only those storage blocks 115 that are different from the base storage image 230.

In a preferred embodiment, the incremental storage images 230 are transmitted to the destination file system 120 with a data structure indicating a set of storage blocks 115 that were deallocated (that is, removed) from the file system on the file server 110. Thus, the images are mark-on-deallocate images of the storage blocks. In response to this data structure, the destination file system 120 removes those indicated storage blocks 115 from its incremental mirror data structure. This allows the destination file system 120 to maintain the incremental mirror data structure at a size no larger than approximately the actual differences between a current file system at the file server 110 and the base storage image 230 from the file server 110.

Consistency Points. When performing either full mirroring or incremental mirroring, it can occur that the transfer of a storage image 230 takes longer than the time needed for the file server 110 to update its consistent file system 114 from a first consistency point to a second consistency point. Consistency points are described in further detail in the WAFL Disclosures.

In a preferred embodiment, the file server 110 does not attempt to create a storage image 230 and to transfer storage blocks 115 for every consistency point. Instead,

after a transfer of a storage image 230, the file server 110 determines the most recent consistency point (or alternatively, determines the next consistency point) as the effective next consistency point. The file server 110 uses the effective next consistency point to determine any incremental storage image 230 for a next transfer.

o Volume Replication. The destination file system 120 can include a disk drive or other accessible storage device. The system 100 can copy storage blocks from the mass storage 113 to that accessible storage device at a signal from the destination file system 120, to provide replicated copies of the file system 114 for updated (read-only) use by other file servers 110.

The file server 110 maintains a set of selected master snapshots 210. A master snapshot 210 is a snapshot 210 whose existence can be known by the destination file system 120, so that the destination file system 120 can be updated with reference to the file system 114 maintained at the file server 110. In a preferred embodiment, each master snapshot 210 is designated by an operator command at the file server 110, and is retained for a relatively long time, such as several months or a year.

In a preferred embodiment, at a minimum, each master snapshot 210 is retained until all known destination file systems 120 have been updated past that master snapshot 210. A master snapshot 210 can be designated as a shadow snapshot 210, but in such cases destination file systems 120 are taken off-line during update of the master

shadow snapshot 210. That is, destination file systems 120 wait for completion of the update of that master shadow snapshot 210 before they are allowed to request an update from that master shadow snapshot 210.

The destination file system 120 generates a message (such as upon command of an operator or in response to initialization or self-test) that it transmits to the file server 110, requesting an update of the file system 114. The message includes a newest master snapshot 210 to which the destination file system 120 has most recently synchronized. The message can also indicate that there is no such newest master snapshot 210.

The file server 110 determines any incremental changes that have occurred to the file system 114 from the newest master snapshot 210 at the destination file system 120 to the newest master snapshot 210 at the file server 110. In response to this determination, the file server 110 determines a storage image 230 including storage blocks 115 for transfer to the destination file system 120, so as to update the copy of the file system 114 at the destination file system 120.

If there is no such newest master snapshot 210, the system 100 performs volume copying for a full copy of the file system 114 represented by the newest master snapshot 210 at the file server 110. Similarly, if the oldest master snapshot 210 at the file

server 110 is newer than the newest master snapshot 210 at the destination file system 120, the system 100 performs volume copying for a full copy of the file system 114.

After volume replication, the destination file system 120 updates its most recent master snapshot 210 to be the most recent master snapshot 210 from the file server 110.

Volume replication is well suited to uploading upgrades to a publicly accessible database, document, or web site. Those destination file systems 120, such as mirror sites, can then obtain the uploaded upgrades periodically, when they are initialized, or upon operator command at the destination file system 120. If the destination file systems 120 are not in communication with the file server 110 for a substantial period of time, when communication is re-established, the destination file systems 120 can perform volume replication with the file server 110 to obtain a substantially up-to-date copy of the file system 114.

In a first preferred embodiment of volume replication (herein called "simple replication"), the destination file system 120 communicates directly (using a direct communication link, a LAN, a WAN, or a combination thereof) with the file server 110.

In a second preferred embodiment of volume replication (herein called "multiple replication"), a first destination file system communicates directly (using a

direct communication link, a LAN, a WAN, or a combination thereof) with a second destination file system. The second destination file system acts like the file server 110 to perform simple replication for the first destination file system.

A sequence of such destination file systems ultimately terminates in a destination file system that communicates directly with the file server 110 and performs simple replication. The sequence of destination file systems thus forms a replication hierarchy, such as in a directed graph or a tree of file severs 110.

In alternative embodiments, the system 100 can also perform one or more combinations of these techniques.

In a preferred embodiment, the file server 110 can maintain a set of pointers to snapshots 210, naming those snapshots 210 and having the property that references to the pointers are functionally equivalent to references to the snapshots 210 themselves. For example, one of the pointers can have a name such as "master," so that the newest master snapshot 210 at the file server 110 can be changed simultaneously for all destination file systems. Thus, all destination file systems can synchronize to the same master snapshot 210.

Shadow Snapshots

The system 100 includes the possibility of designating selected snapshots 210 as "shadow" snapshots 210.

As used herein, a "shadow snapshot" is a subset of a snapshot, the member storage blocks no longer forming a consistent file system. Thus, at one time the member storage blocks of the snapshot did form a consistent file system, but at least some of the member storage blocks have been removed from that snapshot.

A shadow snapshot 210 has the property that the file server 110 can reuse the storage blocks 115 in the snapshot 210 whenever needed. A shadow snapshot 210 can be used as the base of an incremental storage image 230. In such cases, storage blocks 115 might have been removed from the shadow snapshot 210 due to reuse by the file system 110. It thus might occur that the incremental storage image 230 resulting from logically subtraction using the shadow snapshot 210 includes storage blocks 115 that are not strictly necessary (having been removed from the shadow snapshot 210 they are not subtracted out). However, all storage blocks 115 necessary for the incremental storage image 230 will still be included.

For regular snapshots 210, the file server 110 does not reuse the storage blocks 115 in the snapshot 210 until the snapshot 210 is released. Even if the storage

blocks 115 in the snapshot 210 are no longer part of the active file system, the file server 110 retains them without change. Until released, each regular snapshot 210 preserves a consistent file system 114 that can be accessed at a later time.

However, for shadow snapshots 210, the file server 110 can reuse the storage blocks 115 in the shadow snapshot 210. When one of those storage blocks 115 is reused, the file server 110 clears the bit in the shadow snapshot 210 for that storage block 115. Thus, each shadow snapshot 210 represents a set of storage blocks 115 from a consistent file system 114 that have not been changed in the active file system 114 since the shadow snapshot 210 was made. Because storage blocks 115 can be reused, the shadow snapshot 210 does not retain the property of representing a consistent file system 114. However, because the file server 110 can reuse those storage blocks 115, the shadow snapshot 210 does not cause any storage blocks 115 on the mass storage 113 to be permanently occupied.

Method of Operation

Figure 3 shows a process flow diagram of a method for file system image transfer.

A method 300 is performed is performed by the file server 110 and the destination file system 120, and includes a set of flow points and process steps as described herein.

Generality of Operational Technique

In each of the file system image transfer techniques, the method 300 performs three operations:

- o Select a storage image 220, in response to a first file system (or a snapshot thereof) to have an operation performed thereon.
- o Form an image stream 230 in response to the storage image 220. Perform an operation on the image stream 230, such as backup or restore within the first file system, or copying or transfer to a second file system.
- o Reconstruct the first file system (or the snapshot thereof) in response to the image stream 230.

As shown herein, each of these steps is quite general in its application.

In the first (selection) step, the storage image 220 selected can be a complete file system or can be a subset thereof. The subset can be an increment to the complete file system, such as those storage blocks that have been changed, or can be another type of subset. The storage image 220 can be selected a single time, such as for a backup operation, or repeatedly, such as for a mirroring operation. The storage image 220 can be selected in response to a process at a sending file server or at a receiving file server.

For example, as shown herein, the storage image 220 selected can be for a full backup or copying of an entire file system, or can be for incremental backup or incremental mirroring of a file system. The storage image 220 selected can be determined by a sending file server, or can be determined in response to a request by a receiving file server (or set of receiving file servers).

In the second (operational) step, the image stream 230 can be selected so as to optimize the operation. The image stream 230 can be selected and ordered to optimize transfer to different types of media, to optimize transfer rate, or to optimize reliability. In a preferred embodiment, the image stream 230 is optimized to maximize transfer rate from parallel disks in a RAID disk system.

In the third (reconstruction) step, the image stream 230 can be reconstructed into a complete file system, or can be reconstructed into an increment of a file system.

The reconstruction step can be performed immediately or after a delay, can be performed in response to the process that initiated the selection step, or can be performed independently in response to other needs.

Selecting A Storage Image

In each of the file system image transfer techniques, the method 300 selects a storage image 220 to be transferred.

At a flow point 370, the file server 110 is ready to select a storage image 220 for transfer.

At a step 371, the file server 110 forms a logical sum LS of a set of storage images 220 A1 + A2, thus LS = A1 + A2. The logical sum LS can also include any plurality of storage images 220, such as A1 + A2 + A3 + A4, thus for example LS = A1 + A2 + A3 + A4.

At a step 372, the file server 110 determines if the transfer is a full transfer or an incremental transfer. If the transfer is incremental, the method 300 continues with the next step. If the transfer is a full transfer, the method 300 continues with the flow point 380.

At a step 373, the file server 110 forms a logical difference LD of the logical sum LS and a base storage image 220 B, thus LD = LS – B. The base storage image 220 B comprises a snapshot 210.

At a flow point 380, the file server 110 has selected a storage image 230 for transfer.

Volume Copying

At a flow point 310, the file server 110 is ready to perform a volume copying operation.

At a step 311, the file server 111 selects a storage image 220 for transfer, as described with regard to the flow point 370 through the flow point 380. If the volume copying operation is a full volume copy, the storage image 220 selected is for a full transfer. If the volume copying operation is an incremental volume copy, the storage image 220 selected is for an incremental transfer.

At a step 312, the file server 110 determines if the volume is to be copied to disk or to tape.

o If the volume is to be copied to disk, the method 300 continues with the step 313.

- If the volume is to be copied to tape, the method 300 continues with the step 314.

At a step 313, the file server 110 creates an image stream 230 for the selected storage image 220. In a preferred embodiment, the storage blocks 115 in the image stream 230 are ordered for transfer to disk. Each storage block 115 is associated with a VBN (virtual block number) for identification. The method 300 continues with the step 315.

At a step 314, the file server 110 performs the same functions as in the step 313, except that the storage blocks 115 in the image stream 230 are ordered for transfer to tape.

At a step 315, the file server 110 copies the image stream 230 to the destination file system 120 (disk or tape).

- If the image stream 230 is copied to disk, the file server 110 preferably places each storage block 115 in an equivalent position on the target disk(s) as it was on the source disk(s), similar to what would happen on retrieval from tape.

In a preferred embodiment, the file server 110 copies the image stream 230 to the destination file system 120 using a communication protocol known to both the file server 110 and the destination file system 120, such as TCP. As noted herein, the image

stream 230 used with the communication protocol is similar to the image stream 230 used for tape backup, but can include additional messages or packets for acknowledgement or retransmission of data.

The destination file system 120 presents the image stream 230 directly to a restore element, which copies the image stream 230 onto the destination file system 120 target disk(s) as they were on the source disk(s). Because a consistent file system 114 is copied from the file server 110 to the destination file system 120, the storage blocks 115 in the image stream 230 can be used directly as a consistent file system 114 when they arrive at the destination file system 120.

The destination file system 120 might have to alter some inter-block pointers, responsive to the VBN of each storage block 115, if some or all of the target storage blocks 115 are recorded in different physical locations on disk from the source storage blocks 115.

- o If the image stream 230 is copied to tape, the file server 110 preferably places each storage block 115 in a position on the target tape so that it can be retrieved by its VBN. When the storage blocks 115 are eventually retrieved from tape into a disk file server 110, they are preferably placed in equivalent positions on the target disk(s) as they were on the source disk(s).

The destination file system 120 records the image stream 230 directly onto tape, along with a set of block number information for each storage block 115. The destination file system 120 can later retrieve selected storage blocks 115 from tape and place them onto a disk file server 110. Because a consistent file system 114 is copied from the file server 110 to the destination file system 120, the storage blocks 115 in the image stream 230 can be restored directly to disk when later retrieved from tape at the destination file system 120.

The destination file system 120 might have to alter some inter-block pointers, responsive to the VBN of each storage block 115, if some or all of the target storage blocks 115 are retrieved from tape and recorded in different physical locations on disk from the source storage blocks 115. The destination file system 120 recorded this information in header data that it records onto tape.

At a flow point 320, the file server 110 has completed the volume copying operation.

Volume Mirroring

At a flow point 330, the file server 110 is ready to perform a volume mirroring operation.